



Database Systems Project Proposal

Online Language Learning Platform

28.02.2022

Project Group No: 33

Javid Moradi - 21903645

Mustafa Yuşa Babademez - 21703083

Nasuh Dinçer - 21702933

Tarık Buğra Karali -21703937

Course Instructor: Dr. Hamdi Dibeklioglu - Course TA: Zülal
Bingöl

1. Introduction	3
2. Project Description	3
2.1 Why Database System Required?	3
2.2 How Database Will Be Used	4
3. Requirements	4
3.1 Functional Requirements	4
3.1.1 language_natives	4
3.1.2 teachers	4
3.1.3 student	5
3.1.4 new_student	5
3.1.5 admin	5
3.1.6 deadline	5
3.1.7 class	5
3.1.8 subsection	5
3.1.9 assignment	5
3.2 Non-Functional Requirements	6
3.2.1 Quality Requirements	6
3.2.1.1. Usability	6
3.2.1.2. Performance	6
3.2.1.3. Scalability	6
3.2.1.4. Security	7
3.2.1.5. Portability	7
3.2.1.6. Availability	7
3.2.1.6. Capacity	7
3.2.2 Pseudo-Requirements	7
3.2.2.1. Implementation	7
4. Limitations	8
5. Entity-Relationship Diagram	8
6. References	9

1. Introduction

In this report, our database project which is an Online Language Learning Platform will be introduced on different aspects such as requirements, limitations, entity-relationship diagrams. Requirements of the project are examined under two main subtitles which are functional and nonfunctional requirements. The functional requirements represent the functions provided by the system and describe system behavior under specific conditions. On the other hand, non-functional requirements define how the system should behave and focus on the specific business functionalities that an application performs. In the limitation section, some constraints and boundaries of the application are listed. Finally, the E-R model for the Online Language Learning Platform Project represents the conceptual design of the database.

2. Project Description

The aim of this project is to implement an online language learning platform that satisfies all the requirements for teachers, students, language natives, and admin. The system will allow students to request classes according to their levels and create requests for online meetings with natives for speaking exercises. The teachers can give lessons, assign homework to students which they choose, and grade the exams and homework. Additionally, the teachers can see how many students are taking their class and track the students' activities on student accounts. The natives can give speaking exercises and grade them. A native can be a teacher. Lastly, the admin can see the statistics about courses, teachers, and lessons. Admin can see how many people are learning a specific language, a teacher's classes and how many people are in that class, and the average of class grades.

Please find our project's website here:

2.1 Why Database System Required

The Online Language Learning Platform system that we designed requires a database because of the necessity for persistent data storage. The system needs storage to maintain its functionality. The system has different actors such as student, teacher, native, and admin. All actors can perform different actions that cannot be performed without accessing existing data. So that the database system allows us to store and manage data easily. Also, relational database systems provide filtering datasets according to requests and criteria of the user's action and modification of existing data is possible on the database. Additionally, we have different types of users and every type requires different levels of access to data. The database system allows us to configure the access permissions for different types of users. That's why we use databases for storing, accessing, and modifying data. It offers easy and efficient data operation for our application.

2.2 How Database Will Be Used

3. Requirements

3.1 Functional Requirements

3.1.1 language_natives

- Language natives should register the online language learning platform by providing their names, email addresses, passwords, and native languages. Account information is kept for signing in and their native languages separate their field which provides requesting meetings for students.
- Language natives assign online speaking exercises for students who create online meeting requests.
- Language natives grade students' assignments if the type of the assignments is speaking exercise. Otherwise, they are not permitted to grade any kind of assignment.

- Language natives can change their email addresses or their passwords, but they are not allowed to change their native languages since it is assumed that they have a native language even if they know multiple languages.

3.1.2 teachers

- Teachers should register the online language learning platform by providing their names, email addresses, passwords, and language that is taught. Account information is kept for signing in and languages that are taught separate their field which provides requesting lessons for students.
- Teachers teach classes for students who are enrolled in classes. A teacher can teach students who are enrolled in the specified classes which are taught by the teacher.
- Teachers can assign homework for students who have classes to specific teachers. Therefore, a teacher can assign homework for the students who are enrolled in classes that are taught by the teacher.
- Teachers grade students' homework and exams if the type of the assignments is homework or exam. If an assignment is a speaking exercise, they are not permitted to grade the assignment.
- Teachers can see the number of students who enrolled in their classes.
- Teachers can see the activities of students on students' accounts.
- Teachers' reviews are held in *teacher_review*. Comments of students create this attribute.
- Teachers can change their email addresses or their passwords.

3.1.3 student

- Students should register on the online language learning platform by providing their names, email addresses, and passwords. Account information is kept for signing in.
- By picking a language, students can request classes that are taught by teachers according to their levels.

- The levels of students are held in *student_level* which is shaped according to their GPA grades.
- Students' GPA grades are held in *student_gpa* which is calculated according to their grades on their exams and homework.
- Students can request an online meeting request from language natives for speaking exercises. Students can be graded from these exercises by language natives.
- Students can comment on teachers and classes. These comments have effects on *teacher_review* and *class_rating*. Students can only comment on the classes which are enrolled and the teacher of those classes.
- Students' assignments are held in *{student_assignment}* which have type *student_assignment_kind*. Assignments have deadlines.
- Students can change their email addresses or their passwords.

3.1.4 new_student

- New students have *discount_coupon* that is only one-time use and will provide a little cost down upon purchase of any course's price.

3.1.5 admin

- Admins should register on the online language learning platform by providing their names, email addresses, and passwords. Account information is kept for signing in.
- Admins can get student count from teachers in order to analyze how many students are learning a specific language from a teacher.
- Admins get the GPA grades of students to analyze the average grade of the courses.

3.1.6 deadline

- Deadlines have their unique ID, course name, and the date of the deadline.
- Students should complete their assignments before the deadline.

3.1.7 class

- Classes have their unique ID, *class_id*.
- Classes have their own language which is specified according to the teacher.
- Students of classes will be divided according to the level of classes and students. If levels of students and classes match, students can enroll in those classes.
- Classes have a rating (*class_rating*) which is determined by students.
- Classes have a price, *class_price*, which is a mandatory fee to access a course.

3.1.8 subsection

- Subsections have their unique ID, *subsection_id*.
- All subsections must be contained by a class.

3.1.9 assignment

- Assignments have their unique ID, *assignment_id*.
- There are three types of assignments which are exams, homework, and exercises.
- Exams and homework are graded by teachers. Exercises are graded by language natives.

3.2 Non-Functional Requirements

3.2.1 Quality Requirements

3.2.1.1 Usability

- There will be a navigation bar at the top of the page in order to navigate between the pages and to go to the desired page easily. Even if the page changes, this navigation bar will never change to allow users to easily access different pages.
- To assist users who are new to the system, there will be a help page explaining what to do in various situations. When there is a problem in the system, there will be a "Contact Us" section, and users will be able to contact admins directly.
- For certain actions that appear, certain buttons are colored according to the category of the action. For example, when deleting a request, the button will be red to message the user that this is a warning. In addition, for some operations, a

model will appear asking the user if he is sure. In this way, users will make fewer mistakes in the system.

3.2.1.2 Performance

- The designed and implemented website should support 10000 users at a certain time.
- The response of the system to every instruction or order given in the system must be done within 5 seconds at the most.
- Page loading time should be done within a few seconds at most.
- The system must handle the process of storing big data because there will be memory items in the system such as videos and photos that take up too much memory.
- The system should have a high-performance ratio in payment processes, and in this process, performance should be provided without bypassing any security elements.

3.2.1.3 Scalability

- The system should be scalable because the number of students and teachers using our system can always increase. In addition, as this platform reaches more people, the number of employees using our system will increase.

3.2.1.4 Security

- The system must maintain the visibility of passwords at any time because password visibility increases the risk of accounts being stolen.
- The system should not create users until it has verified the password of the user account.
- The system must maintain the visibility of confidential and critical information belonging to users such as nationality ID or phone number.
- The system must be secure against cyberattacks from outside because there will always be students to change their grades illegally.

3.2.1.5 Availability

- The system must be open 24/7 so that users can use the system whenever they need it. In addition, the system should be constantly checked to avoid interruptions in exam or training times.

3.2.1.6 Capacity

- The application will be installed on a database system. Therefore, system teachers should have sufficient capacity to store students and the education system. Initially, the aim is to store 1000 user capacity in the system.

3.2.2 Pseudo-Requirements

3.2.2.1 Implementation

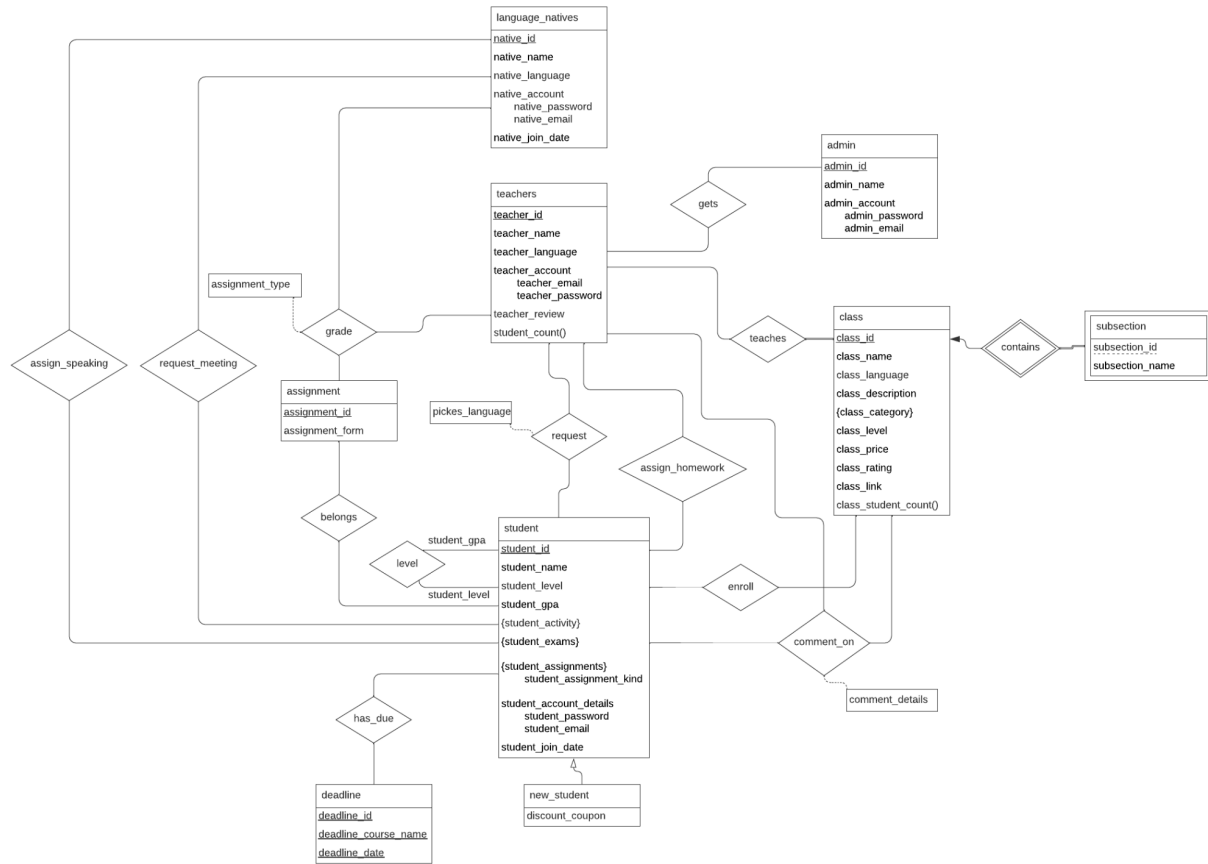
- MySQL will be used as Database Management System.
- React framework will be used for the front-end of the project
- Django will be used for the project's back-end

4. Limitations

- Only admins have the authority to create new courses and assign required instructors to them.
- If the given assignment is not uploaded before the set deadline, the grade of the assignment will be set to zero by the system.
- A user has to have an email or username and password to log in.
- A maximum of one account can be registered with the same email and username for a role.
- Discount offers of new users are one-time use only.
- A student can only rate and comment on his/her registered class only once.

5. Entity-Relationship Diagram

The following is the Entity-Relationship Diagram of the Online Language Learning Platform generated using [1].



For a better view of the diagram, please click [here](#). Note that to access the link active internet connection is required.

6. References

- [1] "Homepage," *Lucid.co*. [Online]. Available: <http://lucid.co>. [Accessed: 27-Feb-2022].